

N85-25873

ISOPARAMETRIC 3-D FINITE ELEMENT MESH GENERATION USING INTERACTIVE COMPUTER GRAPHICS

Cuneyd Kayrak and Dr. Tulga Ozsoy
Department of Mechanical Engineering and Mechanics
Lehigh University

SUMMARY

An isoparametric 3-D finite element mesh generator was developed with direct interface to Lehigh developed interactive geometric modeler program called POLYGON. POLYGON defines the model geometry in terms of boundaries and mesh regions for the mesh generator. The mesh generator controls the mesh flow through the 2-dimensional "spans of regions" by using the topological data defines the connectivity between regions. The program is menu driven and the user has a control of element density and biasing through the spans and can also apply boundary conditions, loads interactively.

INTRODUCTION

The POLYGON (1) package developed at the CAD Laboratory of Lehigh University has been quite successful in providing the user with tools for 3-D wire frames to B-REP type solid model representations conversion process. POLYGON can accept 3-D wire frame data from other CAD systems either directly or through formatted files. POLYGON also provides some geometry generation and manipulation capabilities. Furthermore, different files can be written from POLYGON to be input for other solid modelers use B-REP type representations. The finite element modeling package was developed as a sub module within POLYGON, thus it has a direct access to the wire frame data as well as to the bounded surfaces, "regions" created by POLYGON. The main purpose of this integrated approach is to reach a high degree of automation during the mesh generation by reducing the model preparation time with less required user input. Mesh flow is automatically directed through the number of spans which can be defined using the topological data for regions. Currently only four and/or three-sided regions are allowed. In each region nodes are created using curvi-linear coordinate system and quadratic shape functions (2). Supported element types are linear or parabolic triangular and quadrilateral elements. Loads and restraints can also be applied interactively to the model. Finally, files can be written for input to COSMIC NASTRAN and SDRC's SUPERB finite element analysis programs. Full color visualization of the finite element model is possible at any stage with extensive view control capabilities such as dynamic rotation, scaling, translation, zooming, etc. Interactive menus linked in a tree-like structure allows users a natural way of communication with the system. The program currently has been implemented on DEC's VS11 raster terminals with 16 colors attached to a VAX 11/780, and is supported by Lehigh developed VS11-3D Graphics Package (3).

CURRENT CAPABILITIES OF THE FEM SYSTEM

Finite element model creation consists of six basic steps:

1. Geometry creation: POLYGON provides user with some geometry creation and manipulation capabilities. Points, lines, arcs, conics and cubic splines are currently supported wire frame entity types. Thus, 3-D wire frame geometry can be either created from scratch within POLYGON or on another CAD system and then can be transferred to the POLYGON program. Figure 1 is an example of part-geometry using 3-D wire frame representation.
2. Region creation: Bounded surfaces of planar, ruled and surface of revolution type can be defined automatically and/or manually at this step. The same geometry given in Figure 1 is shown in Figure 2 with all regions defined. These bounded surfaced (polygons), "regions" must be either four or three-sided because of the mesh generation scheme used. Once regions are defined they can be ordered either clockwise or counter-clockwise automatically. The edges of the regions are converted into parabolic type and the region data is formatted in a way so it can be used directly by the FEM module.
3. Span generation: Span defines the flow of mesh generation through the regions connected by opposite edges. Using the edge and region data the program can create spans automatically. Spans can be highlighted for user checking (Figure 3 and Figure 4). Spans can also be created and/or extended manually to include more regions (Figure 5). Number of elements across the span and biasing can be entered by selecting individual spans or system provided default values can be used.

Search for the spans is basically defining the opposite edge of the current region and the other region shares the same edge. Region data contains the edge loop and the edge data contains the region numbers attached to it. A span ends when there is no more region attached to the current edge (free edge) or if the first edge of the span is reached. Each region will be crossed by two spans. After the first span has been defined, search continues to define all other spans crossing all the regions inside the first span. Automatic search ends when all the regions in the model are crossed by two different spans.

4. Mesh generation: Once the spans are defined, mesh is generated throughout the structure at once (Figure 6). First nodes are generated using the number of divisions and biasing information given across the spans. Node generation is performed in parametric space using curvi-linear coordinate system and quadratic shape functions with mid-side nodes placed half-way between corner-nodes, then they are mapped into the current coordinate system. After all the nodes have been defined the elements are generated. Since the nodes are generated in an ordered manner the generation of elements is a trivial process. Nodes and elements are labeled automatically during their generation (Figure 7 and Figure 8). Currently only linear or parabolic triangular and quadrilateral isoparametric element types are supported.

5. Finite element model preparation: For each region different material type numbers can be assigned to the elements. Material property values can be input by the user and stored in material tables which then can be accessed by material type numbers. Default restraint and load definitions can be edited and they can be applied to the selected nodes or along the edges of the region. Different color codings and arrow heads are used for different load types and restraints to enhance the user's visualization.
6. Creating input data for finite element analysis: Once the finite element model information is generated, input files can be written for either COSMIC NASTRAN or SDRC's SUPERB finite element analysis packages.

CONCLUDING REMARKS

The current approach uses the extended concept of regions with span definition. Region refers to a bounded area inside a span and its position is topologically defined. Spans can be referenced as other type of finite element entities but at higher level than regions. Thus a data set assigned to a span can be applied to all the other lower level related entities, i.e. regions, elements, nodes, etc., such as grouping for application of boundary conditions, or blanking for better visualization. By using span definition remarkable savings can be reached easily by reducing the number of user interactions required at model preparation stage. Traditionally all the regions are defined one by one, then the mesh is generated in each of them individually. In Figure 2 the part contains 47 regions but there are only 15 spans. Since the information related to the number of divisions and biasing across the span is entered at span level but not at region level considerable time savings can be reached at this stage of the model preparation process.

REFERENCES

1. Ozsoy, T. and Ochs, J.B., "An Interactive Modeling Program for the generation of Planar Polygons for Boundary Type solids Representations of Wire Frame Models", Proceedings from the Symposium of Computer-Aided Geometry Modeling, NASA Langley, Hampton, VA, April 21, 1983.
2. Zienkiewicz, O.C. and Phillips, D.V., "An Automatic Mesh Generation Scheme for Plane and Curved Surfaces by Isoparametric Co-ordinates," International Journal for Numerical Methods in Engineering, Vol. 3, No. 4, 1971, pp. 519-528.
3. Ozsoy, T. and Ochs, J.B., "Lehigh University's VS11-3D Graphics Package", Proceedings from 1983 DECUS Meeting, St. Louis, MO, May 23, 1983.

ORIGINAL PAGE
COLOR PHOTOGRAPH

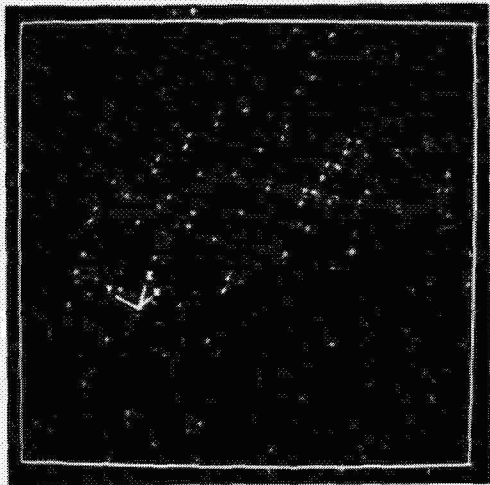


Figure 1. 3-D Wire frame representation of a robot arm with region boundaries

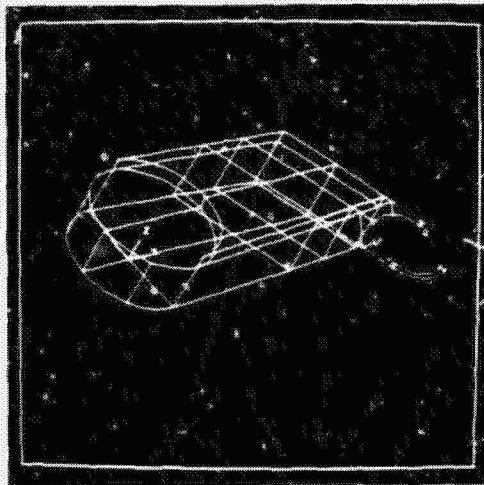


Figure 2. Regions are automatically defined and ordered in clockwise direction.

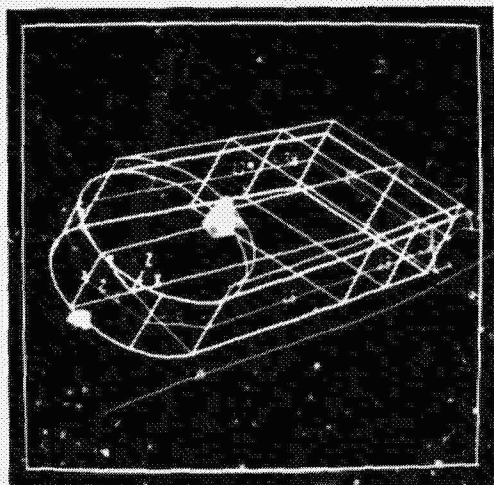


Figure 3. A span is displayed in different color with span direction arrow and region labels on.

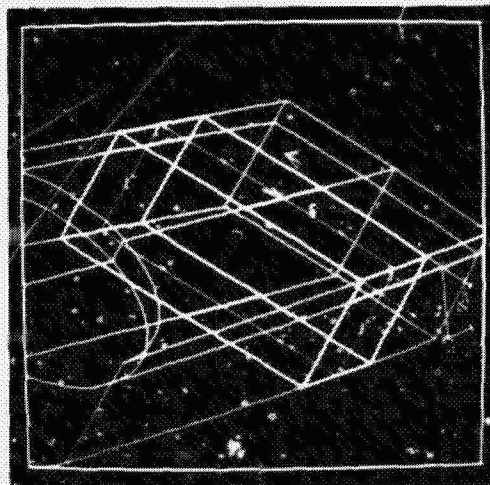


Figure 4. A crossing span is displayed after zoomed in.

ORIGINAL PAGE
COLOR PHOTOGRAPH

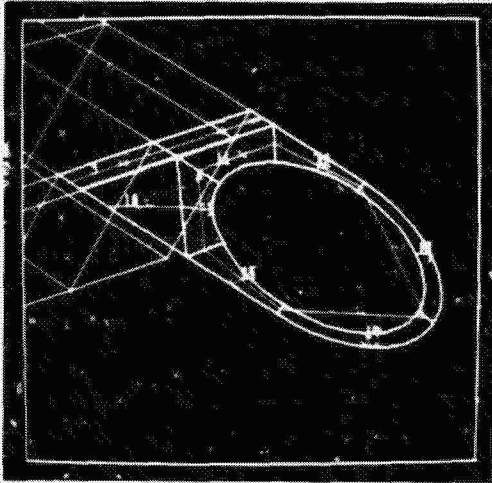


Figure 5. A span is extended manually to include more regions.

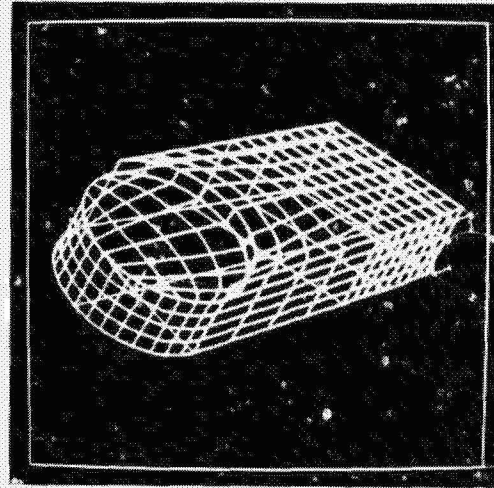


Figure 6. Mesh is displayed in selected regions.

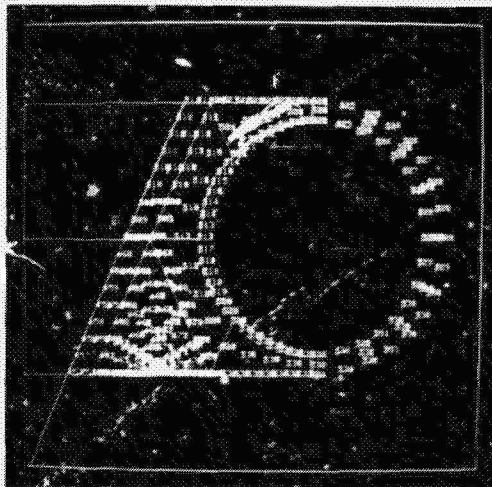


Figure 7. All the nodes in a span are displayed with their labels on.

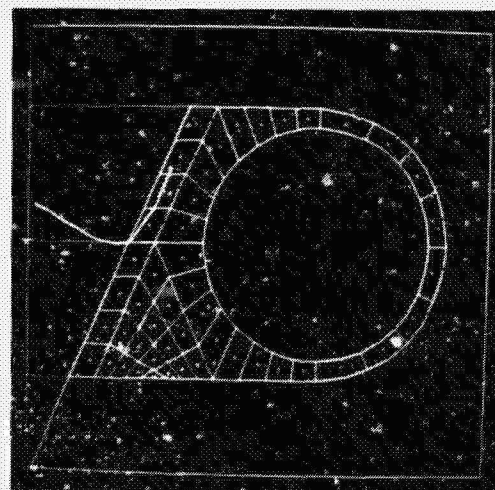


Figure 8. All the elements in a span are displayed with their labels on.